DDM: A Direct Deformation Method of Free Forms for CAD Interface

Juli Yamashita and Yukio Fukui

National Institute of Bioscience and Human-Technology, AIST, MITI
1-1, Higashi, Tsukuba, Ibaraki 305, Japan
E-mail: juli@nibh.go.jp and fukui@nibh.go.jp

[Abstract] A new free form deforming interface named Direct Deformation Method (DDM) is presented in this paper. DDM allows designers to touch and deform free formed surfaces directly, at any point of the surfaces, even if they are represented only by control points and knot vectors. Users do not need to know about such parameters that represent surfaces. The deforming actions of a user are used to calculate new parameters that define the deformed shape of the object. The shape is then reconstructed using the new parameters, allowing the user to visualize his or her deformation. This technique is easy to apply and useful for various form representations.
[Keywords] Direct deformation, CAD, AR, Man-machine interface.

1. Introduction

Today's CAD (Computer Aided Design) interface has two large problems in manipulating tree dimensional free forms. The first, designers should deal free forms with two dimensional (2D) equipment, for example, CRT displays and mice. It is a hard work to recognize 3D free forms on 2D displays. Fortunately, virtual reality (VR) is a promising technology to solve this problem. VR technologies will provide users with realistic truly 3D free form images and I/O devices in the near future.

When the 3D interface devices are improved, the second problem will arise; That is the indirect free form handling method. Most of the existing free form deformation methods are indirect ways. Users can not touch surfaces directly, but some special parameters called control points, weights, and so on. Designers are forced to deal only with those parameters to alter forms they are designing. The more VR technology advances, the more this indirection will cause designers a serious frustration that they can not touch what they see. A method to deform free forms directly is strongly needed.

This paper presents a new direct form deforming technique named Direct Deformation Method (DDM). DDM is a novel interface that allows users to touch and "push" anywhere on surfaces to alter forms, even if they are defined only by control points and knot vectors. It is a simple, but effective technique, that can also be applied to simpler figures, e. g. arcs and polyhedra.

2. Review of Existent Deformation Methods

Existent free form deformation methods can be categorized into tree groups: indirect deformation, simulation, and local geometry manipulation.

(1) Indirect deformation

Most of all deforming methods are indirect; Users can not touch the surface itself, but some special parameters, such as, control points and weights. A surface can be deformed only via those parameters. The Free-Form Deformation [Sede86], function based deformations [Barr84, for example], and implicit surface modeling [Blin82 etc.] can also be categorized in this category. Problems of this deformation type are:

a. Parameters that designers can utilize differ according to surface representations. Users should remember how many kinds of parameters they have on their CAD systems to alter shapes of objects.

b. Deforming effects of parameters differ from each other. It is hard to predict what deformation can be obtained after several parameters' change. To do the inverse is much harder - to determine which parameters and how much change should be made to alter shapes correctly. Designers are forced to do this. They should translate desirable shape change into parameters' change. Users manage to accommodate themselves to this hard work, calculating inverse functions, the translation of desirable shape transformation into parameters' changes. They are expected to be, so to speak, neural inverse function computers! It is useless to waste their creativity on such non essential work.

c. Indirect deformation is unnatural. The more VR technology proceeds, the more natural and direct deformation interface will be needed.

(2) Physically-based simulation

Several deformation methods simulate physical features of real materials, such as, elasticity and volume preservation [e.g. Terz88]. These methods are often used in animations and VR techniques that feedback appropriate forces to users.

From the standpoint of interactive design, these methods have a fatal problem; Calculation time is too long for interactive shape deformation. Physical features are often calculated by FEM (finite element method) that takes much computational time. It is not so much a problem for animations, but it really troubles in interactive interfaces.

Of course, the computational time can be shortened if the granularity of deformation is coarser [Galy91]; they are trade-offs. Today's computers are too slow to provide both enough deformation granularity and quick response interfaces.

In addition, it is doubtful that such physical features as elasticity are indispensable for form design. Design is an essentially inelastic deformation process. Sensory feedback to designers is important, but it does not have to simulate a tactile sensation of real materials.

(3) Local Geometry Manipulation

[Geor91] proposed a new surface deformation method that directly

manipulates surface geometries, such as, tangent vectors, normal vectors, etc., at any point of a curved surface. The method is independent of the surface representation and provides an interactive deformation environment. It must be a good tool for local and minute deformation, however, deformation of wider range is its weak point.

Its another weak point is the gap between designers' image and the mathematical geometries that users can alter. Human visual cognition system has a famous phenomena called "illusion," which plays an important role in form design. For example, a plane is not a plane for our eyes. When we see a geometrically true plane of a single color, we feel as if its center is slightly dented. To avoid giving such a strange impression, designers do not use true planes, but slightly swelled surfaces instead. Similarly, geometrically defined "smoothness," e. g. C^n continuity, does not always congruous with the "smoothness" that we human beings recognize.

Therefore the geometry manipulation technique forces designers to translate their form image into geometrical expressions. It is far from reflecting trails of users' hand motion in moulding forms.

3. Formulating DDM

The basic idea of DDM is that when a cursor (a mouse cursor, a pen, a finger position, etc.) performs a deformation action to shapes (touches and pushes a shape, for example), parameters that represent the shapes are changed automatically so as to deform shapes appropriately to the deforming action. If the deformation is fast enough, the technique allows users to observe the effect of deformation is directly caused by the "pushing action" of the cursor.

The following is a detailed description of DDM. For simplicity, a 2D B-Spline curve of degree 3 is used as an example free form representation, however, DDM can be easily applied to 3D free forms and other shape representations.

This paper assumes that readers have enough knowledge about B-Spline curves and surfaces.

Figure 1 shows the principle of the DDM. The original curve segment C shown in the solid curved line is deformed into C', the dotted curve, according to the following algorithm.

(1) Track a cursor coordinate M(x, y) and check whether it crosses the curve C. If it does, then go to #2 process.

(2) Change parameters that represent C, or control points {V} and its knot vector {t}, to deform the curve according to the crossing motion of the cursor.

$V_i, ..., V_{i+m}$: Corresponding control points of C,

m: An integral number, equal to or more than 0,

$\{t_k, ..., t_{k_1}\}$: Corresponding knot values of C,

m_0: Mouse cursor location before crossing C,

m_1: Mouse cursor location after crossing C,

P: The point where the cursor crosses C,

w_i, ..., w_{i+m}: Weights of V_i, ..., V_{i+m} respectively at P, and

t_p: Knot value at P.

New control points {V'} and new knot vector {t'} are calculated according to a deforming function F as follows:

$$\{\{V'\}, \{t'\}\} = F(\{V_k\}, \{w_k\}, \boldsymbol{m0}, \boldsymbol{m1}, \{t\}, t_p) \qquad \text{(Eq. 1)}$$

where k = i, i+1, ..., i+m.

The only condition that function F must satisfy is:

[Condition 1] C' and vector $\boldsymbol{m_0 m_1}$ do not cross each other, or both coordinates are on the same side of C'.

(3) Display the new curve C' and return to #1.

[Notes]

a. Not only parameter values but numbers of {V'} and {t'} may be different from original ones.

b. Parameter m defines the range that will be deformed by the cursor action.

m = 0 : The deforming effect may be equal to moving the nearest control point to the cursor.

m > degree of C : Wider curve segment will be deformed by one pushing action.

4. A DDM System Implementation

4.1. Implementing DDM

A DDM system, that employs Non-Uniform B-Spline curves of order 4 and Non-Uniform bicubic B-Spline surfaces as its form representation, has been implemented on our IRIS Indigo.

[Interface] "Push" a form with a mouse cursor, then it will be dented smoothly as much as the pushing action (See Figure 2.).

[Implementation] Based on the DDM principle described in chapter 3, new control points {V'} are calculated as follows:

$$V'_j = V_j + a \cdot w_j \cdot (\boldsymbol{m1} - \boldsymbol{m0})$$

where

a : Positive real number and

k = i, i+1, i+2, i+3.

In this system, only control points are changed. Positive number a is determined each time so as to satisfy the condition 1.

4.2. Additional Direct Deformation Primitives

Three additional deformation primitives, which are fixed point, straight line, and angle, are also implemented on the system. These are, however, only for 2D curves now. 3D version is under construction.

(1) Fixed Point

[Interface] Point a point on a form with a mouse cursor. The point becomes fixed when the button is clicked.

[Implementation] First, insert as many knots as degree of the curve at the selected point without changing the form. To make the selected point fixed, fix three control points, the control point coincident with the point and its two neighboring control points, not to be moved by the deformation. Then the selected point becomes free from deforming action and C^2 continuity at the point is guaranteed.

(2) Straight Line

[Interface] Select a part of a curve and push a button. Then the selected part becomes straight (See Figure 3).

[Implementation] Control points that construct the selected part are arranged in a straight line. The straight part can be easily deformed again.

(3) Angle

[Interface] Select a part of a curve and a point on the part. An angle appears when the button is pressed (See Figure 4). The angle and the curve are smoothly connected. Users can alter its position, angle, and lengths of line segments directly through pushing and grabbing.

[Implementation] First, insert as many knots as the degree of the curve at the selected angle point to form a cusp, with which one control point V is coincident. Second, insert one each knot at the two end points of the part. Let V_b and V_e be the nearest control points to the two end points of the selected area. To produce straight line segments around the angle point V, control points between V_{b-1} and V and between V and V_{e+1} are arranged in lines so that V_b and V_e are on them, respectively. Because the curve leaves the control polygon at V_b and V_e, the lengths and the angle formed by straight line segments can be totally controlled by these three control points.

5. Discussion and Future Work

Compared to existing form deformation methods, DDM offers the following merits:

(1) More natural and easier form handling feeling. Shapes are smoothly and rapidly deformed as much as a cursor's push. It is easy to mould the same shape as the trail of designer's hand. The real ability of DDM will be exhibited when it is combined with a 3D input device with force feedback [Fuku92].

(2) Independence from form representations. DDM does not require any knowledge about how shapes are represented mathematically, or, what parameters they have to alter forms. Users can push anywhere on shapes to deform them. It also means that DDM can be applied to many kinds of form representations.

(3) Wide variety of deforming functions. The deforming function F (in Eq. 1) can realize many kinds of deformations, e.g. simple inelastic deformation, elastic deformation, volume preserving deformation, and so on.

(4) DDM does not exclude existing deformation methods. Users can select any technique they like.

DDM also has problems to be solved in the future. Those are:

* What kind of deforming functions should be prepared?
* How to select a set of desirable deforming functions?
* Realization of deformations with topological change. (For example, digging holes and fusing objects together.)
* Combining DDM with 3D I/O device.
* Evaluation of DDM's usability through psycho-physical experiments.

The first three problems can be solved by "tool metaphor," the system will prepare special tools for such deformations that change topology of forms. Users can select appropriate deformation tools. "Gesture" will also add much to improve DDM interface.

6. Summary

This paper presented a new direct form deformation method DDM. With DDM, users can deform free forms by "pushing" them directly with a cursor; Then the shape deformed as much as the "push." Because DDM is independent from form representation, designers do not need to know such parameters as control points. DDM also provides many kinds of deforming functions.

In the near future, DDM will be combined with the 3D input device with force feedback that Dr. Fukui, one of the authors, is developing and increase "touch and feel" in 3D form deforming process.

References

[Barr84] Barr, A. H.: "Global and Local Deformations of Solid Primitives", in Proc. of SIGGRAPH '84, 1984.

[Blin82] Blinn, J. F.: "A Generalization of Algebraic Surface Drawing", ACM Transactions on Graphics, Vol.1, No.3, 1982.

[Fuku92] Fukui, Y. and M. Shimojo: "Edge Tracing of Virtual Shape Using Input Device with Force Feedback," Trans. of the Inst. of Electronics, Info. and Comm. Engineers, Vol. J74–D–II, No. 8, 1992. (In Japanese)

[Galy91] Galyean, T. A. and J. F. Hughes: "Sculpting: An Interactive Volumetric Modeling Technique," in Proc. of SIGGRAPH '91, 1991.

[Geor92] Georgiades, P. N. and D. P. Greenberg: "Locally Manipulating the Geometry of Curved Surfaces", IEEE Computer Graphics & Applications, 1992.

[Sede86] Sederberg, T. W. and S. R. Parry: "Free-Form Deformation of Solid Geometric Models", in Proc. of SIGGRAPH '86, 1986.

[Terz88] Terzopoulos, D. and K. Fleischer: "Deformable Models", The Visual Computer, 14, Springer-Verlag, 1988.

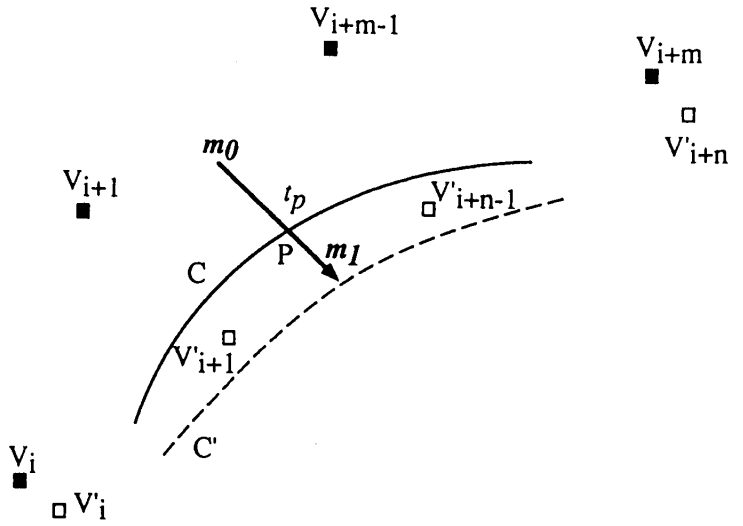

Figure 1.
A cursor crosses a B-Spline curve segment C shown in a solid line, which is deformed into C', the dotted curve. The deformation is realized by creating new appropriate curve difining parameters {V'} and {t'} automatically. Note that not only the values but the numbers of new parameters may be different from the original ones.

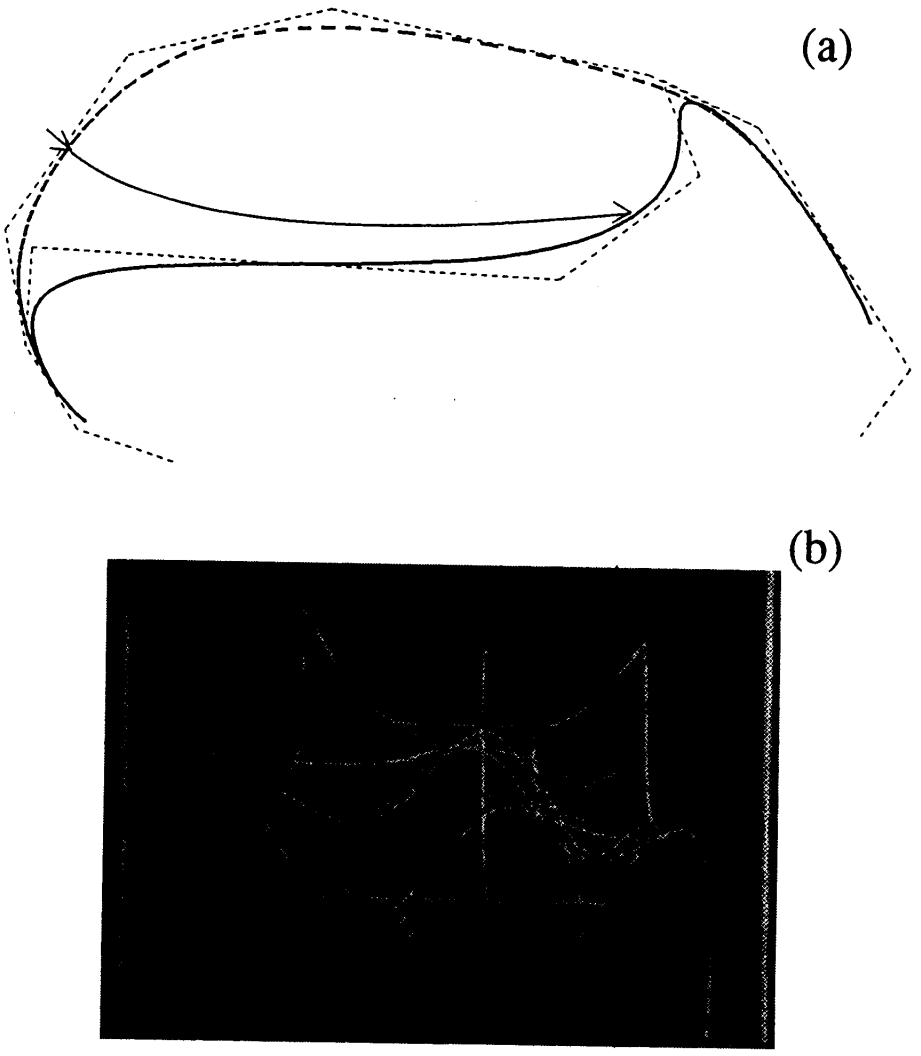

Figure 2.
Effect of DDM.
(a) Thick dotted line is the original curve, which is deformed into the thick solid curve with DDM. Thin dotted lines are control polygons. The thin solid line shows the mouse cursor's trail. The original curve is "pushed" by the cursor and deformed. Note that several control points are moved by one "pushing" action simultaneously.
(b) 3D bicubic B-Spline surface deformation with DDM.

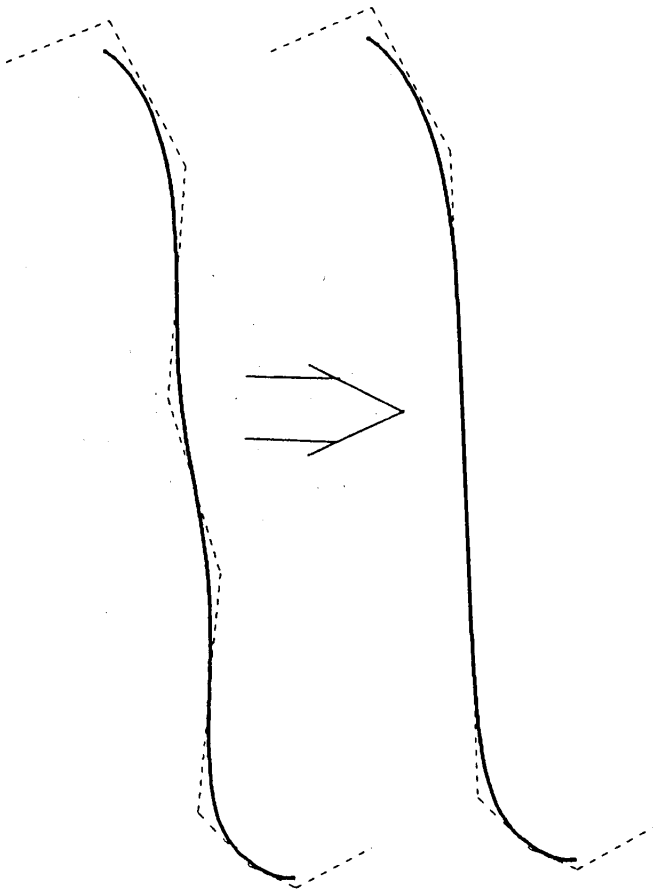

Figure 3.
Straight line. To produce a straight line, corresponding control points are arranged to form a line.

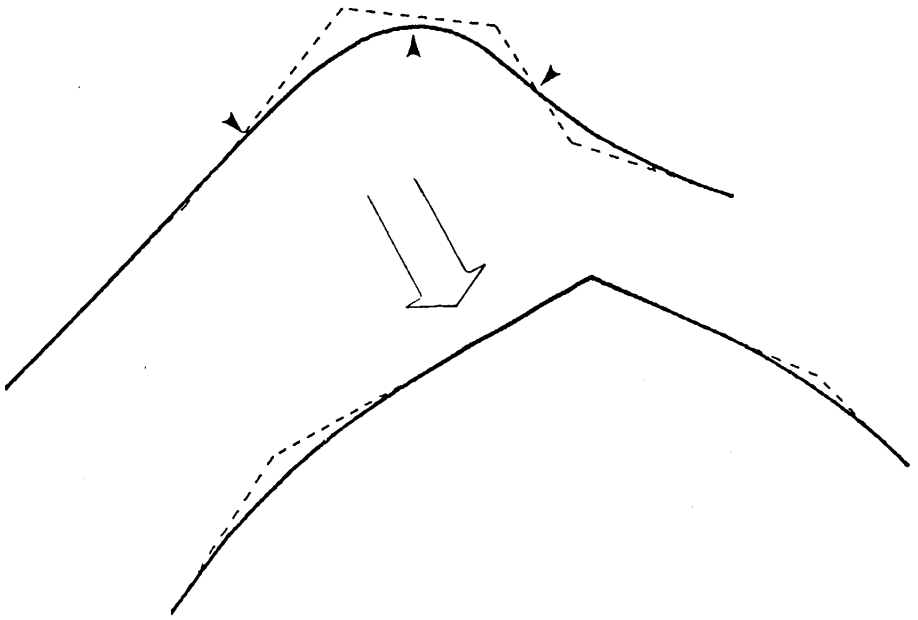

Figure 4.
Angle. Lengths and the angle of line segments and position can be controlled by a mouse cursor. Thin dotted lines are control polygons.